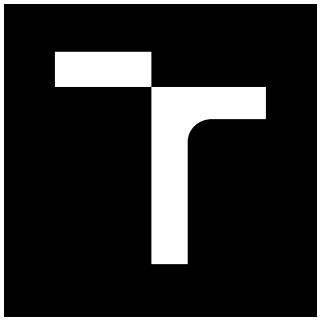
VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

BAKALÁŘSKÁ PRÁCE

Brno, 2017 Tomáš Koutný



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV TELEKOMUNIKACÍ
DEPARTMENT OF TELECOMMUNICATIONS

BEZPEČNÁ AUTENTIZACE UŽIVATELŮ POMOCÍ ČIPOVÝCH KARET
SECURE USER AUTHENTICATION USING SMART CARDS

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE — **Tomáš Koutný**
AUTHOR

VEDOUCÍ PRÁCE — **Ing. Lukáš Malina, Ph.D.**
SUPERVISOR

BRNO 2017

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

Bakalářská práce

bakalářský studijní obor **Teleinformatika**
Ústav telekomunikací

Student: Tomáš Koutný ***ID:*** 164748
Ročník: 3 ***Akademický rok:*** 2016/17

NÁZEV TÉMATU:

Bezpečná autentizace uživatelů pomocí čipových karet

POKYNY PRO VYPRACOVÁNÍ:

Analyzujte a porovnejte čipové karty Basic Cards z hlediska nabízených kryptografických funkcí a míry zabezpečení. Otestujte jednotlivé kryptografické operace a změřte jejich dobu výpočtu na čipové kartě.
Navrhněte bezpečný autentizační protokol a řešení implementujte na čipové karty typu Basic Cards. Ověřte funkčnost navrženého řešení a zhodnoťte paměťovou a výpočetní výkonost řešení i celkovou bezpečnost autentizačního protokolu.

DOPORUČENÁ LITERATURA:

[1] Menezes, Alfred, Van Oorshot, Paul, Vanstone, Scott. Handbook of applied cryptography. Boca Raton : CRC Press, 1997. 780 s. ISBN 0849385237.

[2] Guilfoyle, Tony. The ZeitControl BasicCard Family. 2012. 320 s.

Termín zadání: 1.2.2017 ***Termín odevzdání:*** 8.6.2017

Vedoucí práce: Ing. Lukáš Malina, Ph.D.
Konzultant:

doc. Ing. Jiří Mišurec, CSc.
předseda oborové rady

UPOZORNĚNÍ:
Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

ABSTRAKT

Tato práce se věnuje problematice čipových karet typu BasicCard a jejich analýzou z hlediska nabízených kryptografických funkcí a míry zabezpečení. Byla zde použita metoda eliptických křivek. Práce také obsahuje návrh autentizačního protokolu a jeho implementaci.

KLÍČOVÁ SLOVA

Chytrá karta, Basic karta, Šifrování, Eliptické křivky, Autentizace

ABSTRACT

This thesis deals with Basic Card problém and their analysis in terms of the offered features cryptographic security measures. Elliptic curve method was used in this thesis. The thesis contains design of authentication protocol and its implementation.

KEYWORDS

Smart card, Basic card, Encryption, Elliptic curve, Authentication

KOUTNÝ, Tomáš Bezpečná autentizace uživatelů pomocí čipových karet: bakalářská práce. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, 2017. 33 s. Vedoucí práce byl Ing. Lukáš Malina, Ph.D.

Prohlášení

Prohlašuji, že svou bakalářskou práci na téma „Bezpečná autentizace uživatelů pomocí čipových karet“ jsem vypracoval samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této bakalářské práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom následků porušení ustanovení S 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno……. Podpis autora…….

Poděkování

Rád bych poděkoval vedoucímu diplomové práce panu Ing. Lukáši Malinovi, Ph.D. za odborné vedení, konzultace, trpělivost a podnětné návrhy k práci.

Brno…….

Podpis autora…….

Obsah

Seznam obrázků

Seznam tabulek

Úvod

Tato práce se věnuje problematice čipových karet a různým kryptografickým metodám, které se u nich používají. Chytré karty jsou dnes velice rozšířené a každý se s nimi už setkal a nejspíš i nějakou vlastní. Nejrozšířenější typ karty, který se řadí do čipových karet, jsou karty SIM, které se používají v mobilních telefonech. Další velice rozšířenou podobou čipové karty jsou debetní nebo kreditní karty, kterými platíme v obchodech.

V této práci se nejprve seznámíme s tím, co to vlastně je čipová karta a jak vypadá. Zjistíme, že se karty dělí na paměťové a mikroprocesorové a vysvětlíme si jaký je mezi nimi rozdíl. Dále bude zmínka o způsobu komunikace mezi kartou a čtečkou karet, jelikož existují karty kontaktní a bezkontaktní. Zaměříme se i na protokoly komunikace.

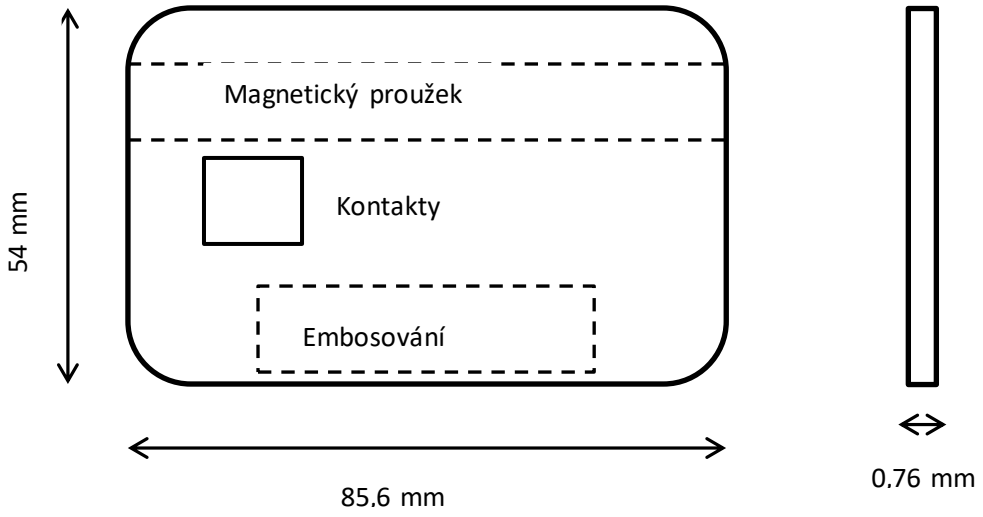
Dále si řekneme něco o BasicCard. Zjistíme, jaké jsou podporovány kryptografické algoritmy a jaké typy BasicCard vlastně existují. Dozvíme se, co je to kryptografie na bázi eliptických křivek a základní metody pro práci s nimi v programovacím jazyce Basic.

V praktické části bylo úkolem analyzovat a porovnat čipové karty Basic Card z hlediska nabízených kryptografických funkcí a míry zabezpečení. Dále otestovat jednotlivé kryptografické operace a změřit dobu jejich výpočtu. Nakonec bylo cílem této práce implementovat bezpečný autentizační protokol na Basic Card. Praktická část také obsahuje změřené výsledky a popis navrženého autentizačního protokolu. Program, který je také součástí této práce byl testován na kartě typu ZC7.5 REV D.

1. Teoretická část

1.1 Čipové karty

Čipové karty jsou dnes velice rozšířené a každý se s nimi už určitě setkal. Používají se například jako cestovní doklad, platební karty nebo SIM karty, ale používají se i pro přístup do střežených prostor. Existují karty různých velikostí, ale nejčastěji se setkáme s kartami typu ID-1(např. platební karty), jejichž rozměr je 85,60 × 53,98 mm (3,370 × 2,125 palce) s tloušťkou 0,76 mm. Tyto rozměry určuje mezinárodní norma ISO/IEC 7816. Další je například typ ID-000. Tento typ karet je spíš známý jako SIM karta a jeho rozměry jsou 25 mm × 15 mm x 0,76 mm, ale v dnešní době už jsou velice rozšířené i tzv. micro SIM a nano SIM. Dodržování těchto rozměrů je důležité hlavně u dotykových karet, jako jsou např. právě zmíněné SIM karty nebo platební karty. U bezdotykových karet jsou rozměry libovolné a neplatí tam téměř žádná omezení.



Obr. 1.1: Popis karty

1.1.1 Rozdělení čipových karet

Čipové karty můžeme dělit podle dvou základních kritérií. Zaprvé se karty dělí na kontaktní a bezkontaktní. Kontaktní karty se používají nejčastěji jako platební karty

nebo jako SIM karty. Bezkontaktní karty se používají zejména v systémech kontroly a evidence vstupů, docházkových systémech, a systémech pro odbavení cestujících v hromadné dopravě. Zadruhé se karty dělí podle toho, jestli má karta pouze paměť nebo jestli obsahuje i vlastní mikroprocesor.

Paměťové karty

Nejvíce používaná a nejlevnější karta je paměťová karta. Tento typ karet obsahuje ROM a EEPROM (Electrically Erasable Programmable Read-Only Memory) paměti. Tyto paměti jsou nevolatilní, což znamená, že po odpojení karty od zdroje napětí (např. čtečka paměťových karet), karta udrží informace, které obsahuje. Velikost pamětí je v řádech několika kilobajtů. V tomto typu karet není žádný mikroprocesor, takže je nelze přeprogramovat. Zabezpečení paměťových karet se většinou provádí blokováním přístupu k určitým oblastem paměti pomocí bezpečnostních klíčů nebo lze provádět pouze určité operace s pamětí. Nevýhoda je, že se dá tento druh karet celkem snadno padělat. Existují tři základní typy paměťových karet: Přímá (straight memory card), chráněná/segmentovaná (protected/segmented memory card) a karta uchovávající hodnotu (stored value memory card).

Přímá karta (Straight memory card)

Tyto karty pouze uchovávají data a nemají žádné schopnosti, jak data spravovat. Také se nedokáží sami identifikovat čtečce, takže systém musí vědět, jaký typ karty byl vložen do čtečky. Tyto karty se dají snadno duplikovat.

Chráněná karta (Protected/segmented memory card)

Tento druh karet má v sobě zabudovanou kontrolu přístupu do paměti karty. Někdy je proto nazývána jako inteligentní paměťová karta. Některé tyto karty můžou být nakonfigurovány tak, aby omezovali zapisování nebo čtení z karty. Obvykle se k tomu využívá hesla nebo systémového klíče. Není lehké ji duplikovat, ale stále existují možnosti, jak zabezpečení obejít.

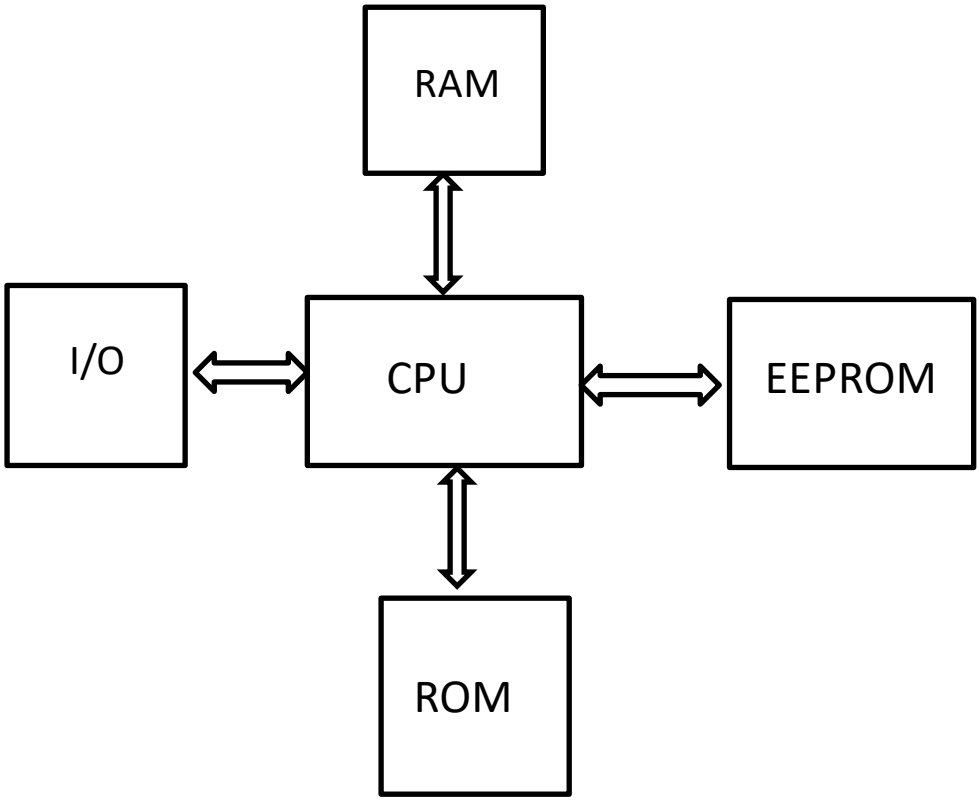
Karta uchovávající hodnotu (Stored value memory card)

Tyto karty jsou vytvořeny pro účel uchovávání hodnot. U většiny těchto karet se zabezpečení vytváří už při výrobě. Zabezpečení tvoří například heslo nebo se využívá nějaké zašifrované logické operace. Na kartě je velice malá nebo žádná volná paměť

pro další funkce. Typicky se jedná o SIM karty, které po provolání určitého obnosu přestanou fungovat.

Mikroprocesorové karty

Mikroprocesorové karty mají oproti paměťovým kartám navíc procesor, díky kterému je možné tyto karty programovat a je zajištěna i vyšší bezpečnost a také obsahují RAM paměť. Další výhoda je, že jsou multifunkční, takže se jedna karta dá používat pro více činností. U těchto typů karet je možné přistoupit k datům jen přes zabudovaný mikroprocesor. Není žádný jiný způsob, jak přistoupit k datové oblasti, protože není žádné spojení mezi kontakty na kartě a samotnou pamětí. Vložený mikroprocesor umožňuje také chránit data na kartě pomocí kryptografických metod před neoprávněným přístupem. Paměťové čipové karty nabízí celou řadu bezpečnostních služeb, jako je autentizace, šifrování, digitální podpis, které mohou být použity v důvěryhodném prostředí. Kromě symetrické kryptografie (AES, DES) obsahují karty i asymetrické šifrování (RSA) nebo generátory náhodných čísel pro lepší ochranu před útokem.

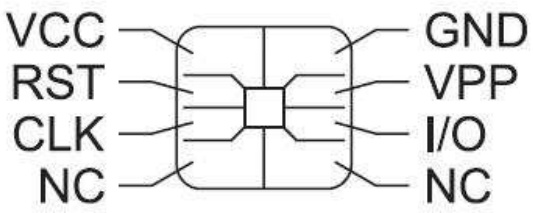


Obr. 1.2: Schéma mikroprocesorové karty

Procesor (CPU) je obvykle 8-bitový nebo 16-bitový. Paměť ROM bývá 12-30kb a je v ní uložen operační systém, komunikační protokoly a algoritmy pro komunikaci. Operační systém je odpovědný za zabezpečení dat, protože přístup k paměti řídí právě operační systém. Tuto paměť nelze přepsat a inicializuje se v průběhu výrobního procesu. V EEPROM paměti se zaznamenávají údaje aplikací na kartě a také bezpečnostní klíče. Vzhledem k tomu, co EEPROM obsahuje je přístup k této paměti chráněn. RAM paměť po odpojení napájení ztrácí svůj obsah. Některé karty mají i kryptografický modul, který se zabývá zabezpečením. Karty s tímto modulem jsou dražší, než bez kryptografického modulu stejně jako jsou mikroprocesorové karty dražší než paměťové karty.

Kontaktní a bezkontaktní karty

V závislosti na komunikaci chytré karty se čtečkou se karty dělí na kontaktní a bezkontaktní. Kontaktní karty komunikují se čtečkou pomocí 8 pozlacených kontaktů. Pomocí těchto pinů je i karta během komunikace napájena, takže karta neobsahuje žádnou baterii. Komunikace mezi kartou a čtečkou u kontaktních karet je založena na normě ISO 7816 T=0. Pro komunikaci je použit protokol APDU. Každý pin má svůj význam. Po připojení karty do čtečky se vytvoří komunikačních okruh mezi kartou a čtečkou. Komunikace probíhá v režimu half-duplex. Half-duplex (česky polo-duplex) je režim střídavé obousměrné komunikace. V daném okamžiku může probíhat přenos pouze v jednom směru. Směr přenosu se ale může měnit.



Obr. 1.3: Kontakty na kartě

VCC	Napájení karty
RST	Resetování komunikace
CLK	Hodinový signál
GND	Uzemnění
VPP	Napětí potřebné pro programování
I/O	Sériová komunikace
NC	nevyužito

Tab. 1.1: Seznam pinů

Bezkontaktní čipová karta nemá vyvedený kontakt, proto musí čtečka a karta obsahovat antény, pomocí kterých mezi sebou komunikují. Čipová karta je napájena pomocí elektromagnetického vlnění, které vysílá čtečka. Nemůže tedy fungovat, aniž by byla v dosahu čtečky. Komunikace probíhá většinou ve vzdálenosti 10 cm a méně, ale jsou i karty s větším dosahem kolem 1 metru. Karty komunikují pomocí bezdrátové technologie. Používání bezkontaktních karet je samozřejmě jednoduší, jelikož si nemusíme dávat pozor na orientaci karty. Také odpadá riziko nefunkčnosti komunikace mezi kartou a čtečkou z důvodu poškození pinů, které se opětovným zasouváním do čtečky opotřebovávají. Bezkontaktní karty jsou dražší než karty kontaktní.

Dnes již existují i tzv. hybridní karty. Hybridní čipová karta obsahuje dva nezávisle čipy. První čip komunikuje s okolím pomocí vestavěné antény. Druhý čip komunikuje pomocí kontaktů na kartě. Tyto čipy fungují nezávisle na sobě a nejsou nijak propojené. Čtečky chytrých karet jsou většinou k počítači připojeny pomocí sériového portu nebo pomocí USB.

1.1.2 Výroba čipových karet

Na cestě ke konečnému výrobku je několik kroků, kterými musí karta projít. Nejprve je výroba těla karty, což zahrnuje výrobu plastu, tisku a dalších prvků jako je například magnetický proužek. Následuje vložení modulu čipové karty a proces dokončení a inicializace. Všechny čipy procházejí testováním, které má zajistit, aby se k zákazníkovi dostali jen kvalitní produkty. Optická a elektrická personalizace udělá z karty jedinečnou.

Základní materiál pro karty je dodáván jako fólie nebo jako granulát. Klasicky používaným materiálem je PVC, ale kvůli ochraně životního prostředí se začínají karty vyrábět i z jiných materiálů.

Pro tisk se nejvíce využívá tzv. technika ofsetového tisku. Proces tisku začíná výrobou ofsetových tiskových desek pro každou barvu tisku, obvykle přímo z digitálního zdroje vytvořeného počítačem.

Tělo karty je tvořeno několika fóliemi, které jsou zahřáty a lisovány k sobě. Tyto fólie jsou vyrobeny z termoplastu. Nejčastěji se karta skládá ze 4 až 5 vrstev, ale u některých druhů karet může být těchto vrstev až 9. Bez ohledu na to, kolik vrstev je použito, šířka karty musí být vždy maximálně 0,84 mm.

Dalším krokem je umístění modulu do těla čipové karty. Pro modul je v kartě připravena prohlubeň, do které se vloží. Čip je následně předpersonalizován programem určeným k danému typu karty.

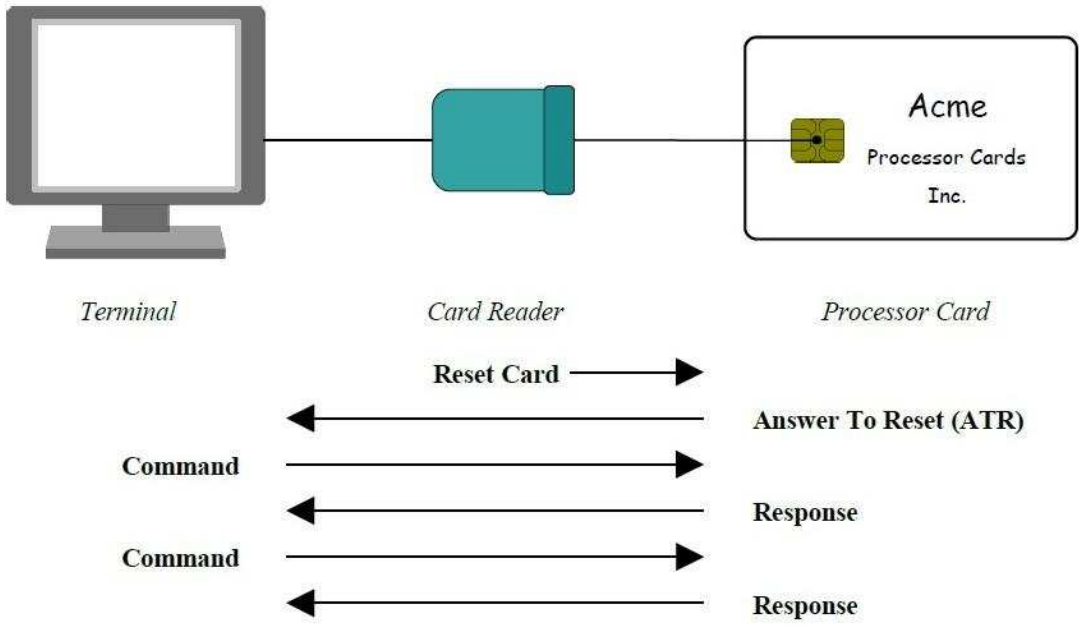
1.1.3 Čtecí zařízení

Čtečky čipových karet jsou někdy označovány také jako programátory karet (mohou na kartu zapisovat) nebo jako terminály karet. Mezi čtečkou a terminálem karet je ale rozdíl. Termín čtečka se obecně používá k popisu jednotky, která je propojena s PC pro většinu svých požadavků na zpracování. Naproti tomu terminál je samostatné zařízení pro zpracování.

Smart karty jsou přenosné datové karty, které musí komunikovat s jiným zařízením, aby získaly přístup k zobrazovacímu zařízení nebo síti. Karty mohou být připojeny do čtečky nebo mohou pracovat s rádiovými frekvencemi. Když čipová karta a čtečka karet přicházejí do kontaktu, každá strana se identifikuje tím, že zasílá a přijímá informace. Pokud vyměněné zprávy neodpovídají, nedochází k dalšímu zpracování.

1.1.4 Komunikace karty se čtečkou

Komunikace karty a čtečky probíhá v režimu master - slave. Master - slave je architektura, ve které jedno zařízení (master) řídí provoz jednoho nebo více jiných zařízení (slaves). V tomto případě je master čtečka a karta je slave. Karta nikdy nepošle odpověď bez předchozí výzvy. Když kartu vložíme do čtečky, karta vykoná reset a pošle čtečce zprávu o resetu ATR (answer to reset). Čtečka vyhodnotí tuto zprávu a odešle kartě příkaz, co má udělat.



Obr. 1.4: Komunikace karty se čtečkou, převzato z[8]

1.1.5 Protokoly komunikace

Pro komunikaci s kartou existuje celá řada protokolů. V komunikačních protokolech jsou implementovány příkazy a odpovědi pro případ, že nastane chyba při komunikaci. Dva nejrozšířenější jsou protokoly T=0 a T=1. Oba tyto protokoly jsou mezinárodní.

Protokol T=0 byl první mezinárodně standardizovaný protokol pro chytré karty. Byl vytvořen v dobách, kdy se chytré karty teprve začínaly využívat a tak je designovaný tak, aby využíval minimum paměti a je také znám svou jednoduchostí. Protokol T=0 je half – duplex (polo - duplex) a bajtově orientovaný. Nejmenší zpracovávanou jednotkou je tedy jeden bajt. Tento způsob přenosu po bajtech je sice jednoduchý, ale není moc bezpečný. Komunikační jednotka je tvořena z hlavičky, která obsahuje Class byte, command byte a tři parametry, následuje datová část.

Protokol T=1 je asynchronní, poloduplexní protokol pro chytré karty. Je založen na mezinárodním standardu ISO/IEC 7816-3. T=1 protokol je blokově orientovaný, což znamená, že jeden blok je nejmenší jednotka, která může být přenesena mezi kartou a terminálem. Tento protokol může být přiřazen k datové vrstvě v referenčním modelu OSI. T=1 je v současnosti jediný mezinárodní protokol pro chytré karty, který dovoluje

všechny druhy zabezpečení. Sekvence komunikace začíná ve chvíli, kdy karta pošle ATR po úspěšném provedení PPS (protocol parameter selection).

Přenášené bloky jsou používány ke dvěma základním účelům. První je transparentní přenos dat, které specifikují aplikaci a druhý účel je posílání kontrolních dat. Přenášený blok se skládá z počátečního pole (prologue field), informačního pole (information field) a z koncového pole (final epilogue field). Počáteční pole a koncové pole jsou povinná a vždy musí být poslána. Pole obsahující informaci je nepovinné a obsahuje data pro aplikační vrstvu. Existují tři odlišné typy bloků v tomto protokolu: informační bloky, bloky potvrzující příjem a systémové bloky. Informační bloky se používají pro transparentní výměnu dat na aplikační vrstvě. Bloky potvrzující odpovědi, které neobsahují žádná datová pole, se používají pro pozitivní nebo negativní potvrzení přijetí. Systémové bloky obsahují kontrolní informace.

U bezkontaktních karet se používá T=CL (Contactless) protokol. Tento protokol je definován mezinárodním standardem ISO/IEC 14443. Protokol T=CL je podobný protokolu T=1. Je definován jako sekvence zpráv, které si vymění PCD (čtečka bezkontaktních karet) a PICC(karty). Komunikace začne tzv. antikolizní smyčkou (anticollision loop), kterou vyšle čtečka ve chvíli kdy zachytí ve své blízkosti kartu. Tato smyčka se postará o to, že i když je v dosahu čtečky více karet, začne komunikovat s tou správnou. Následně čtečka přijme zprávu ATS. Po přijmutí zprávy ATS čtečka pošle kartě APDU obsahující příkaz, který má karta udělat. Karta poté odešle APDU s odpovědí.

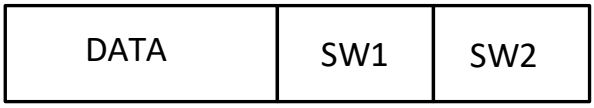
1.1.6 APDU

APDU (application protocol data unit) je datová jednotka aplikačního protokolu, který probíhá mezi čipovou kartou a čtečkou. Struktura APDU je definována normou ISO/IEC 7816-4 .

APDU mají dvě kategorie: příkaz APDU a odpověď APDU. Příkaz APDU je posílán čtečkou směrem do karty – obsahuje povinnou 4 - bytovou hlavičku (CLA, INS, P1, P2) a data o velikosti 0 až 255 bytů. Odpověď APDU je poslána z karty do čtečky - obsahuje povinný 2 - bytový status a data o velikosti 0 až 256 bytů.

CLA	INS	P1	P2	L_c	Data příkazu	L_e

Obr. 1.5: Příkaz APDU



Obr. 1.6: Odpověď APDU

Příkaz APDU		
Název pole	Velikost (B)	Popis
CLA	1	Instrukční třída – určuje typ příkazu
INS	1	Instrukční kód – určuje konkrétní příkaz
P1-P2	2	instrukční parametry příkazu
L_C	0,1 nebo 3	Šifruje (N_c) bytů následujících dat příkazu
Data příkazu	N_c	N_c bytů dat
L_e	0, 1, 2 nebo 3	Šifruje maximálně (N_e) bytů očekávané odpovědi

Tab. 1.2: Příkaz APDU

Odpověď APDU		
Název pole	Velikost (B)	Popis
Data odpovědi	N_r (max. N_e)	Data odpovědi
SW1-SW2	2	Status provedení příkazu

Tab. 1.3: Odpověď APDU

1.1.7 Operační systém čipové karty

Čipová karta je řízena kódem, který implementuje příkazy komunikačního rozhraní ISO APDU a spravuje data na kartě. Tento kód se nazývá operační systém čipové karty. Na tomto operačním systému, který je implementován výrobcem při výrobě karty závisí, k čemu je čipovou kartu možné použít a pomocí jakých příkazů jsou dostupné požadované funkce. OS je uložen v ROM (Read Only Memory) paměti karty. Jelikož

karty mají omezenou paměť a výkon, lze na kartách používat jen základní datové typy a základní funkce.

1.2 Basic karty

Basic karty jsou čipové karty, které lze programovat pomocí programovacího jazyka Basic. Existují tři druhy basic karet: rozšířené (enhanced), profesionální (professional) a víceaplikační (multiaplication). Tyto karty obsahují následující:

Komunikační protokoly:

- T=0 komunikace na úrovni bajtů definovaná v ISO/IEC 7816-3: Elektronické signály a přenosové protokoly
- T=1 komunikace na úrovni bloků definovaná v ISO/IEC 7816-3: Elektronické signály a přenosové protokoly
- T=CL Typ A bezkontaktní protokol definován v ISO/IEC 14443: Proximity karty
- Mifare bezkontaktní protokol NXP polovodičů

Kryptografické algoritmy:

- RSA algoritmus veřejných klíčů se 4096-bitovým klíčem
- EC-p algoritmus eliptických křivek s velikostí pole 544b
- EC-167 a EC-211 binární algoritmus eliptických křivek
- DES šifrovací standard s 8-, 16-, 24-bytovým klíčem
- AES pokročilý šifrovací standard
- Hash algoritmy SHA-1, SHA-224, SHA-256, SHA-384, a SHA-512

1.2.1 Typy basic karet

Rozšířené basic karty						
Verze	PK Algoritmus	EEPROM	RAM	Protokol	Šifrování	Hash
ZC3.12, ZC3.13	EC-161	2K	256bytů	T=1	DES, AES	SHA-1
ZC3.32, ZC3.33	EC-161	8K	256bytů	T=1	DES, AES	SHA-1
ZC3.42, ZC3.43	EC-161	16K	256bytů	T=1	DES, AES	SHA-1

Tab. 1.4: Rozšířené basic karty

Profesionální basic karty						
Verze	PK Algoritmus	EEPROM	RAM	Protokol	Šifrování	Hash
ZC5.4	EC-167, EC-211	16K	2K	0, 1	D, A, E, O	SHA-256
ZC5.5	EC-167, EC-211	32K	2K	0, 1	D, A, E, O	SHA-256
ZC5.6	EC-167, EC-211	60K	2K	0, 1	D, A, E, O	SHA-256
ZC7.4	All PK	16K	4.3K	0, 1, CL, M	D, A, E, O, SM	All SHA
ZC7.5	All PK	32K	4.3K	0, 1, CL, M	D, A, E, O, SM	All SHA
ZC7.6	All PK	72K	4.3K	0, 1, CL, M	D, A, E, O, SM	All SHA

Tab. 1.5: Profesionální basic karty

Víceaplikační basic karty						
Verze	PK Algoritmus	EEPROM	RAM	Protokol	Šifrování	Hash
ZC6.5	EC-167, EC-211	31K	1,7K	0, 1	D, A, E, O	SHA-256
ZC8.4	All PK	16K	4,3K	0, 1, CL, M	D, A, E, O, SM	All SHA
ZC8.5	All PK	32K	4,3K	0, 1, CL, M	D, A, E, O, SM	All SHA
ZC8.6	All PK	72K	4,3K	0, 1, CL, M	D, A, E, O, SM	All SHA

Tab. 1.6: Multiaplikační basic karty

All PK - RSA, EC-167, EC-211, EC-p

All SHA - SHA-1, SHA-224, SHA-256, SHA-384, SHA-512

1.2.2 Jazyk ZC-Basic

Jazyk ZC-Basic byl vytvořen hlavně pro basic karty. Může ale běžet i na počítači, s nebo bez čtečky, která je připojená sériovým portem nebo pomocí USB. ZC-Basic program, který běží na počítači, bude obvykle komunikovat s jedním nebo více programy běžícími na kartě. Kompilátor může vytvořit spustitelný soubor, obraz souboru a debug soubor z terminálového zdrojového programu.

Spustitelný soubor

Kompilátor může vytvořit spustitelný soubor (executable file) s příponou .exe, který poběží jako konzolová aplikace ve windows. Program také může komunikovat s reálnou nebo simulovanou basic kartou. Takto spuštěné programy nemůžou zapisovat do EEPROM paměti.

Obraz souboru

Pro větší flexibilitu během vývoje programu může kompilátor vytvořit obraz souboru (image file) s příponou .IMG ze zdrojového terminálového programu. Terminálový program může běžet současně s programem běžícím na basic kartě.

Debug soubor

Kompilátor může také vytvořit debug soubor s příponou .DBG, ve kterém jsou zahrnuty záznamy o chybách. Tyto soubory využívá terminálový debugger.

Terminálový soubor programu

Debugger terminálového programu uloží data do terminálového souboru (terminal program file) s příponou .ZCT. Tento soubor obsahuje například jméno zdrojového souboru, nastavení kompilátoru a jiná data.

1.2.3 Systémové knihovny

Funkčnost ZC-Basic jazyka může být rozšířena používáním systémových knihoven. V terminálových programech a profesionálních a multiaplikačních kartách jsou systémové knihovny implementovány jako knihovny pluginů, které jsou načteny do EEPROM paměti, když jsou potřeba. Novou knihovnu lze přidat pomocí:

#Include library.def

Tyto knihovny jsou nezbytné pro deklarování funkcí, které obsahují. V současnosti jsou dostupné tyto knihovny:

Název	Terminal	Enhanced Basic Card	Professional Basic Card	Multi-Application Basic Card
RSA	●		*	
EC-p	●		*	
EC-211	●		*	●
EC-167	●		*	●
EC-161	●	●		
Component	●			●
TMLib	●		*	
Crypto	●		*	
BigInt	●		*	
AES	●	●	*	●
EAX	●		*	●
OMAC	●		*	●
SHA	●	●	●	●
TLVLib	●		*	
Mifare			*	*
MATH	●			
MISC	●	●	●	●

Tab. 1.7: Dostupné knihovny

*Tyto systémové knihovny nejsou dostupné na všech Basic kartách

RSA-Rivest-Shamir-Adleman knihovna

RSA knihovna implementuje Rivest-Shamir-Adleman kryptografii veřejných klíčů. Tato knihovna podporuje následující operace:

- Generování veřejného a soukromého klíče
- Šifrování a dešifrování
- Generování a ověření digitálního podpisu

Princip RSA je následující:

- Zvolí se dvě různá náhodná prvočísla *p* a *q*
- Spočítá se jejich součin *n=p*q*
- Spočítá se hodnota Eulerovy funkce $\varphi(n) = (p - 1) * (q - 1)$
- Dále se vygeneruje celé číslo *e*, které je větší než 1, ale menší než $\varphi(n)$ a je s $\varphi(n)$ nesoudělné
- Nalezneme číslo *d*, pro které platí $de \equiv 1 \pmod{\varphi(n)}$
- Zašifrování zprávy se provede $c = m^e \bmod n$

Knihovna binárních eliptických křivek

Binární eliptické křivky jsou eliptické křivky nad binárním polem GF(2^n). Enhanced basic card, profesionální karty série ZC5 a multiaplikační karty obsahují knihovny binárních eliptických křivek.

Jsou dostupné tři knihovny:

- Knihovna EC-211 nad polem GF(2^{211}) s 211-bitovým klíčem. Odpovídá ekvivalentu 2048-bitového klíče u RSA.
- Knihovna EC-167 nad polem GF(2^{167}) se 167-bitovým klíčem. Odpovídá ekvivalentu 1024-bitového klíče u RSA.
- Knihovna EC-161 nad polem GF(2^{168}) se 161-bitovým klíčem. Tato knihovna je dostupná u všech Enhanced Basic karet.

Všechny tyto knihovny podporují následující operace:

- Generování soukromého a veřejného klíče
- Generování session klíče
- Generování digitálního podpisu
- Ověření digitálního podpisu (tato funkce není podporována u Enhanced Basic karet)

BigInt knihovna

Systémová knihovna BigInt implementuje aritmetické operace pro velké integery, které jsou reprezentovány proměnnými typu string. Tato knihovna je dostupná pro karty série ZC7 od verze REV C. Knihovnu BigInt lze přidat pomocí:

#Include BigInt.def

Maximální velikost integeru závisí na maximální možné velikosti stringu, což je 2048 bytů (16384 bitů) u karet série ZC7 a 16384 bytů (131072 bitů) u terminálové aplikace. Avšak velikost vstupních parametrů je většinou limitována hardwarem.

Většina procedur v knihovně má 2 varianty. První je *Funkce*, která vrací výsledek jako string a druhá je varianta je Předpis(Sub), který vrací výsledek přepsáním prvního parametru. Předpis by se měl použít v případě, že už první parametr nebude dále potřebný.

BigInt knihovna umožňuje následující aritmetické operace:

BigIntCompare	porovná *x* a *y*
BigIntAdd	*x+y*
BigIntSub	*x-y*
BigIntMul	*x*y*
BigIntDiv	*x/y*
BigIntRem	*x Mod y*
BigIntDivRemInPlace	spočítá *x/y* a *x Mod y*
BigIntAnd	*x And y*
BigIntOr	*x Or y*
BigIntXor	*x Xor y*
BigIntPower	x^e mod n

AES knihovna

Tato knihovna implementuje pokročilý šifrovací standard. AES používá Rijndael algoritmus jako kryptografický základ. Standard specifikuje tři povolené velikosti klíčů: 128-bitový, 192-bitový a 256-bitový. Všechny tyto velikosti klíčů jsou dostupné v terminálových programech a jsou podporovány v profesionálních kartách série ZC5 a v multiaplikačních kartách série ZC6. K načtení této knihovny se použije příkaz:

#Include AES.DEF

SHA knihovna

Knihovna SHA (The secure hash algorithm library) implementuje tyto algoritmy:

- SHA-1 s délkou hashe 20 bajtů, tento algotritmus je dostupný na všech verzích karet
- SHA-224 s délkou hashe 28 bajtů, tento algotritmus je dostupný na kartách série ZC7 od verze REV C
- SHA-256 s délkou 32 bajtů, tento algotritmus je dostupný na kartách série ZC5, ZC6 a ZC7
- SHA-384 s délkou hashe 48 bajtů, tento algotritmus je dostupný na kartách série ZC7 od verze REV C
- SHA-512 s délkou hashe 64 bajtů, tento algotritmus je dostupný na kartách série ZC7 od verze REV C

OMAC knihovna

Tato knihovna slouží pro autentizaci zpráv. Je dostupná v terminálových programech, u profesionálních karet série ZC5 a ZC7 a u multiaplikačních karet série ZC6. K načtení této knihovny se použije příkaz:

#Include OMAC.DEF

Algoritmus spočítá 16-bajtový tag, který identifikuje zprávu. Pro vypočítání tagu se používá tato funkce:

Function OMAC (Type%, Key$, Mess$) As String

parametr *Type%* je délka klíče, která musí být 128, 192 nebo 256 bitů.

1.2.4 Kryptografie basic karet

Jak již bylo zmíněno, na basic kartách lze použít několik šifrovacích algoritmů s různou mírou zabezpečení. Do těchto algoritmů patří například RSA, DES, kryptografie na bázi eliptických křivek nebo například různé hashovací algoritmy. Dnešní systémy, které

používají veřejný klíč, byly vytvořeny proto, aby byl vyřešen problém klíčového hospodářství. Pro N účastníků komunikačního procesu potřebujeme (N*(N - 1))/2 tajných klíčů a to přináší problémy například s distribucí klíčů. Některé problémy odpadají v případě, že účastníci komunikace používají dvojici klíčů. Jeden veřejný a druhý soukromý. Nevýhodou těchto asymetrických kryptosystémů je, že jsou pomalejší než systémy symetrické.

1.2.5 Kryptografie na bázi eliptických křivek

Kryptosystémy eliptických křivek jsou založeny na problému diskrétního logaritmu (Discrete logarithm problem - DLP). Velikost eliptické křivky určuje složitost problému. Předpokládá se, že stejné úrovně zabezpečení, jakou nabízejí RSA systémy s velkým modulem, lze dosáhnout s podstatně menší skupinou eliptických křivek. Eliptické křivky jsou algebraické struktury definované nad konečnými tělesy, která lze algebraicky klasifikovat a každé konečné těleso je určeno svým řádem. Eliptická křivka je hladká spojitá křivka, na které definujeme bod $\mathcal{O}$, což je bod v nekonečnu.

Obecná rovnice přímky má tvar:

$$ax + by + c = 0 \tag{1.1}$$

kde a ≠ 0 nebo b ≠ 0 a x, y jsou souřadnice libovolného bodu přímky. Eliptická křivka je definována pomocí Weierstrassovy rovnice:

$$E: y^2 + a_1xy + a_2y = x^3 + a_3x^2 + a_4x + a_5 \tag{1.2}$$

Častěji se však setkáme se zjednodušeným zápisem Weiestrassovy rovnice ve formátu:

$$y^2 = x^3 + ax + b \tag{1.3}$$

Koeficienty a, b musí splňovat:

$$4a^3 + 27b^2 \neq 0 \tag{1.4}$$

V případě, že by koeficienty nesplňovaly tuto podmínku, nejednalo by se o eliptickou křivku. Souřadnice bodů na křivce x, y jsou reálná čísla.

Existují dva typy eliptických křivek, které se používají v kryptografii: křivky nad primárním polem (prime field) pro prvočíslo p; a křivky nad binárním polem pro celá čísla. Křivky nad prime field jsou implementovány v EC-p knihovně a jsou dostupné pro karty verzí SC7 a SC8. Křivky nad binárním polem jsou implementovány ve většině ostatních karet. Karty, které používají metodu eliptických křivek, vyžadují koprocesor k udržení rychlosti odpovědi.

EC-p knihovna podporuje následující operace:

- Generování soukromého a veřejného klíče
- Generování session klíče
- Generování digitálního podpisu
- Ověření digitálního podpisu
- Násobení a sčítání bodů na křivce

Generování klíče

V knihovně EC-p je klíč celé číslo velikosti integeru. Veřejný klíč má dva formáty: expanded formát ve formě (x, y), kde x a y jsou kladná čísla; compressed formát ve formě (x, ỹ). Pro generování soukromého a veřejného klíče slouží metoda:

Call ECpGenerateKeyPair (PrK$, PuK$)

Soukromý klíč je vracen pomocí PrK$ a veřejný klíč je vracen pomocí PuK$.

Generování digitálního podpisu

Jsou dostupné dvě varianty digitálního podpisu v EC-p. Nyberg-Rueppel (NR) a Digital signature algorithm (DSA). Není mezi nimi téměř žádný rozdíl. Rozdíl je v příkazu, kterým se tyto podpisy generují. Pro NR slouží příkaz:

Signature$ = ECpHashAndSignNR (PrivateKey$, Data$)

Pro DSA slouží příkaz:

Signature$ = ECpHashAndSignDSA (PrivateKey$, Data$)

Ověření těchto podpisů se následně dělá pomocí:

Status = ECpHashAndVerifyNR (Signature$, Data$, PublicKey$)
Status = ECpHashAndVerifyDSA (Signature$, Data$, PublicKey$)

Násobení a sčítání bodů na křivce

Série karet ZC7 a ZC8 od verze REV D podporují jednoduché sčítání a násobení bodů na eliptické křivce.

Pro sčítání slouží:

ECpAddPoints (P$, Q$)

a pro násobení:

ECpMultiplyPoint (P$, n$)

1.2.6 Dostupné křivky

V knihovně EC-p je předdefinováno 19 druhů křivek s různými velikostmi polí. Tyto velikosti se pohybují od 160 bitů do 521 bitů.

Index křivky	Velikost[b]	ID křivky
1	160	brainpoolP160r1
2	160	brainpoolP160t1
3	192	brainpoolP192r1
4	192	brainpoolP192t1
5	224	brainpoolP224r1
6	224	brainpoolP224t1
7	256	brainpoolP256r1
8	256	brainpoolP256t1
9	320	brainpoolP320r1
10	320	brainpoolP320t1
11	384	brainpoolP384r1
12	384	brainpoolP384t1
13	512	brainpoolP512r1
14	512	brainpoolP512t1
15	192	P-192
16	224	P-224
17	256	P-256
18	384	P-384
19	521	P-521

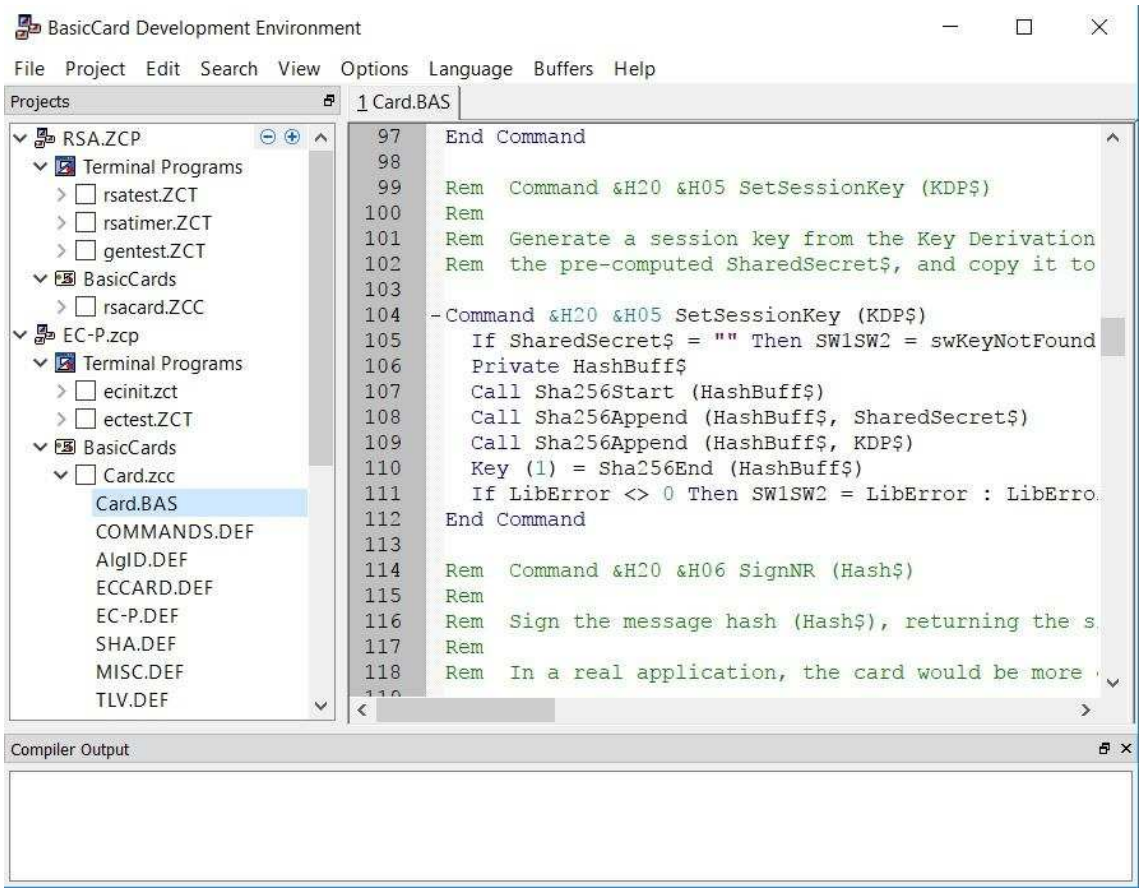
Tab. 1.8: Dostupné křivky

Křivky 1-14 jsou Brainpool Standard Curves a jsou definovány pro karty série ZC7. Velikost polí těchto křivek je od 160 do 512 bitů. Křivky 15-19 jsou NIST Recommended Elliptic Curves a jsou pro karty série ZC7 od REV C. Velikost polí u těchto křivek je od 192 do 521 bitů.

2. Praktická část

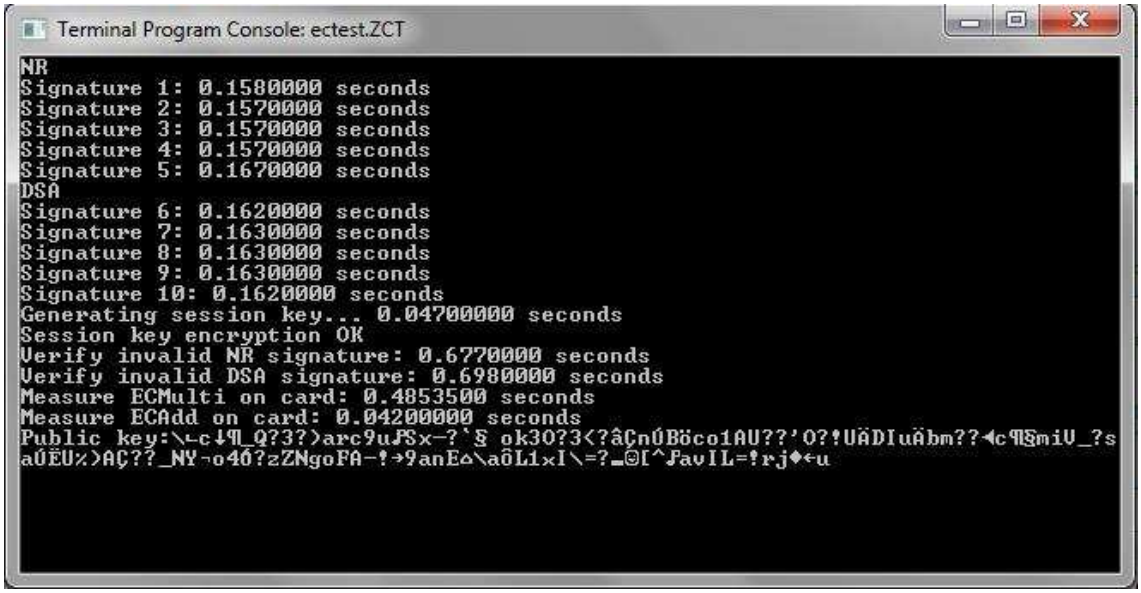
V praktické části bylo úkolem seznámit se s čipovými kartami typu BasicCard. Tento typ karet porovnat z hlediska nabízených kryptografických funkcí a otestovat jednotlivé kryptografické operace a změřit jejich dobu výpočtu na čipové kartě.

Vývojové prostředí, ve kterém lze programovat chytré karty nese název BasicCard Development Enviroment. Jak již název tohoto programu napovídá, toto prostředí slouží pro programování karet v jazyce basic. Po vytvoření programu se udělá několik souborů, nás ale zajímají především soubory s příponou .bas, ve kterých definujeme jednotlivé operace a metody.



Obr. 2.1: Vývojové prostředí

Po spuštění programu se otevře jedno či více oken příkazového řádku, ve kterém se zobrazí výstup naší aplikace.



Obr. 2.2: Ukázka výstupu programu

Testování probíhalo na kartě druhu ZC7.5 typu REV D. Tato karta podporuje různé druhy šifrovacích algoritmů a disponuje 32K EEPROM paměti. Pro otestování jednotlivých kryptografických operací jsem použil metodu šifrování na bázi eliptických křivek. Konkrétně se jednalo o operace ověření pomocí Nyberg-Rueppel (NR) signature a Digital signature algorithm (DSA) a dále pak násobení a sčítání bodů na eliptických křivkách. Všechno tyto operace jsem testoval na všech dostupných druzích křivek, které jsem již dříve zmínil.

Curve ID	NR Signature[ms]	DSA Signature[ms]
brainpoolP160r1	231	230
brainpoolP160t1	230	231
brainpoolP192r1	270	266
brainpoolP192t1	266	262
brainpoolP224r1	302	301
brainpoolP224t1	297	301
brainpoolP256r1	339	346
brainpoolP256t1	331	344
brainpoolP320r1	407	415
brainpoolP320t1	406	407
brainpoolP384r1	478	486
brainpoolP384t1	479	488
brainpoolP512r1	678	690
brainpoolP512t1	675	680
P-192	259	267
P-224	308	299
P-256	338	341
P-384	493	488
P-521	709	717

Tab. 2.1: Změřené výsledky (NR a DSA)

Curve ID	ECMulti[ms]	ECAdd[ms]
brainpoolP160r1	160,8	23
brainpoolP160t1	162,7	23,1
brainpoolP192r1	192,8	23,9
brainpoolP192t1	185,9	24
brainpoolP224r1	225,3	25,3
brainpoolP224t1	216,2	25,3
brainpoolP256r1	244,3	27,3
brainpoolP256t1	250,9	27,1
brainpoolP320r1	310,9	32,7
brainpoolP320t1	308,4	32,8
brainpoolP384r1	377	37,7
brainpoolP384t1	367,8	37,8
brainpoolP512r1	494,1	42,1
brainpoolP512t1	479	42
P-192	193,8	23,9
P-224	218,8	25,2
P-256	253	27,1
P-384	367,1	37,7
P-521	510,6	45,6

Tab. 2.2: Změřené výsledky (Multi a Add)

Výsledky měření můžete vidět v tabulkách změřených hodnot. V prvním sloupci je ID křivky, kde číslo vyjadřuje velikost v bitech. V následujících sloupcích je vždy název operace (NR signature, DSA signature, násobení a sčítání) a změřený čas, jak dlouho trval výpočet dané operace na kartě. Čas je vždy uveden v milisekundách. Pro dosažení větší přesnosti měření, jsem program pro každou křivku spustil 20x a následně zprůměroval naměřené hodnoty. Ze změřených výsledků je patrné, že se stoupající velikostí křivky se zvyšuje i čas potřebný pro výpočet jednotlivých operací na kartě.

2.1 Implementace autentizačního protokolu

Dalším úkolem bakalářské práce bylo implementovat již navržený autentizační protokol, převzatý z [4].

2.2 Popis autentizačního protokolu

Před samotným spuštěním autentizačního procesu musí mít obě strany (uživatel a ověřovatel) ustanoveny veřejné parametry kryptosystému *g, p, q, H()*. Každý uživatel vypočítá unikátní veřejný klíč $ID\_i = g^{key\_i} \bmod p$. Soukromý klíč key_i je náhodně vygenerován. Tyto klíče jsou uloženy na čipové kartě a klíč ID_i je uložen i v databázi ověřovatele. Ověřovatel je v tomto případě myšlena čtečka karet. Pro generování klíčů je v knihovně EC-p speciální funkce *ECpGenerateKeyPair*, která vygeneruje náhodný privátní a veřejný klíč. Pro spočítání ID_i použijeme funkci z knihovny BigInt, konkrétně se jedná o funkci *BigIntPower*.

Dalším krokem je zahájení autentizačního protokolu výpočtem kryptografického závazku $c = g^{r} \bmod p$. To se provede stejně jako výpočet veřejného klíče ID_i. Dále uživatel odešle veřejný klíč *ID_i* k ověřovateli. Komunikace mezi kartou a čtečkou se realizuje tak, že například do paměti karty uložím nějaký string pomocí:

EEPROM MyData as String

následně určím co bude obsahovat:

Command &H80 &H00 GetData(LC=0, Data as String)
Data=MyData
End Command

Command &H80 &H02 SetData(Data as String, Disable Le)
MyData=Data
End Command

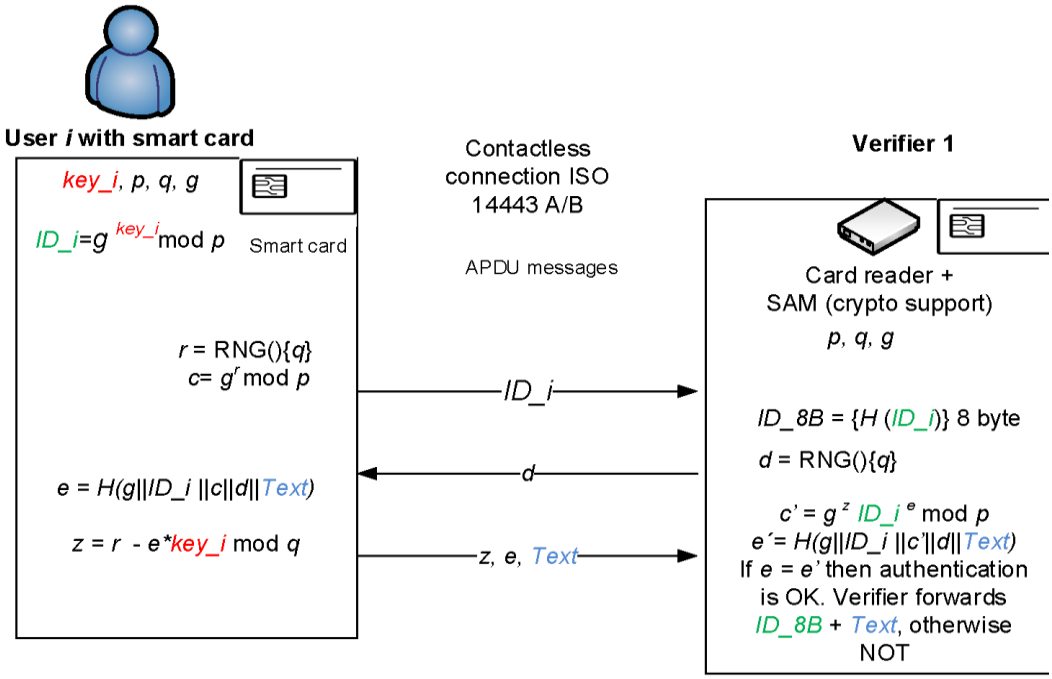
takto vytvořený string můžu následně zavolat pomocí:

Public Data as String
Call GetData(Data) : CheckSW1SW2

Ověřovatel vygeneruje náhodné číslo d a pošle ho zpět uživateli. Náhodná čísla lze generovat více způsoby. V tomto případě, jelikož má být náhodné číslo 40, 128 nebo 160-bitové, použijeme příkaz *Call RandomString(d$, 16)*. Tento příkaz zapříčiní to, že se vygeneruje náhodný 16-bajtový (128-bitový) string. Dále ověřovatel vypočítá pomocí hash funkce a funkce ořezaní dat zkrácený identifikátor uživatele *ID_8B*.

Uživatel ověří, že *d* je nenulové a je menší než *q*. Dále vypočítá otisk dat *e* jako hash $H(g|ID\_i||c||d||Text)$. Pole *Text* je nepovinné a je využito pouze v případě potřeby přenosu dat nebo atributů. Dále uživatel vypočítá odpověď $z = r - e*key\_i \bmod q$. Parametry *z, e* a případný *Text* odešle uživatel k ověřovateli.

Ověřovatel přijme parametry *z, e* a případný *Text* a vypočítá kryptografický závazek $c' = g^{z} ID\_i^{e} \bmod p$. Následně ze získaných dat od uživatele vypočítá otisk dat e' a porovná tuto hodnotu s přijatým otiskem dat *e* od uživatele. V případě rovnosti těchto hodnot je autentizace vyhodnocena jako úspěšná. V případě nerovnosti hodnot je autentizace vyhodnocena jako neúspěšná.



Obr. 2.3: Specifikace navrženého autentizačního protokolu, převzato z [4]

Definice parametrů (verze 1024 bitů):

g, p, q – jsou sdílené veřejné parametry kryptosystému, kde p je prvočíselný modulus o velikosti 1024 bitů, g je základ diskrétního logaritmu o velikosti větší než 1 a menší než 1024 bitů a q je prvočíselný dělitel čísla p-1 o velikosti 160 bitů

key_i – je soukromý klíč o velikosti 160 bitů, který je uložen pouze na kartě během registrace uživatele

ID_i – je identifikátor uživatele o velikosti 1024 bitů, který je i veřejným klíčem uživatele

ID_8B – je zkrácený identifikátor uživatele z ID_i. Tento zkrácený identifikátor o velikosti 64 bitů je uchovaný v databázi pro ověření přístupových práv.

r – představuje náhodné číslo, nenulové a menší než q

c – je kryptografický závazek o velikosti p k číslu r, tzv. svědek

d – je tzv. výzva od ověřovatele, tj. náhodné číslo, nenulové a menší než q. Typicky d může být 40 bitů, 128 bitů a max. 160 bitů.

e – je tzv. otisk dat na straně uživatele

z – je tzv. odpověď o velikosti q, která je vypočtena na základě znalosti soukromého klíče uživatele key_i, otisku dat e a náhodného čísla r

c‘– je obnovený kryptografický závazek na straně ověřovatele o velikosti p

e‘– je obnovený otisk dat na straně ověřovatele o velikosti q

H() – je vhodná hashovací funkce, např. SHA-256 s ořezaným výstupem na 20 B, případně SHA-1 s výstupem 160 bitů / 20 bajtů

3. Závěr

V rámci této bakalářské práce jsem se seznámil s čipovými kartami typu Basic Card. Analyzoval a porovnal jsem čipové karty z hlediska nabízených kryptografických funkcí. Dále jsem otestoval jednotlivé kryptografické operace a změřil dobu jejich výpočtu na kartě.

Základ praktické části této práce tvoří program vytvořený ve vývojovém prostředí BasicCard Development Environment v jazyce basic. Tento program řeší kryptografii na bázi eliptických křivek a byla v něm prováděna měření časů potřebných pro výpočet operací na kartě. Také je zde možnost měnit velikosti polí eliptických křivek a to konkrétně od 160 do 512 bitů pro Brainpool Standard Curves a od 192 do 521 bitů pro NIST Recommended Elliptic Curves.

Díky tomuto programu jsem zjistil, že velikost eliptické křivky se projeví v době, která je potřebná pro výpočet daných operací na kartě docela významně. Pro křivku s velikostí pole 160 bitů trvalo ověření podpisu pomocí DSA 230ms, zatímco u křivky s polem o velikosti 512 bitů toto ověření trvalo 690ms. Z důvodu zpřesnění naměřených hodnot, jsem spustil program 20x a následně změřené hodnoty zprůměroval.

Dále bylo cílem implementovat navržený autentizační protokol. Tento autentizační protokol se dá označit za bezpečný, jelikož uživatel musí prokázat znalost soukromého klíče, který je uložen pouze v jeho čipové kartě. Pokud tento soukromý klíč uživatel nezná, nelze spočítat správnou odpověď na výzvu a ověřovatel zamítne uživateli přístup. Pro zvýšení bezpečnosti lze využít kód PIN, díky kterému po několika neúspěšných pokusech o zadání klíče systém zablokuje přístup uživateli. Samotná implementace se zdařila jen z části. Jednotlivé klíče se mi sice podařilo vygenerovat nebo spočítat, následně se mi už ale nepodařilo tyto klíče přesunout na kartu. Program tedy obsahuje porovnání kryptografických operací a měření doby jejich výpočtu. A dále obsahuje jednotlivé výpočty veřejných a soukromých klíčů a generování náhodných čísel potřebných k implementaci autentizačního protokolu.

Literatura

[1] EBNER, Claus. Smart card production environment [online]
Dostupné z:
https://www.google.cz/url?sa=t&rct=j&q=&esrc=s&source=web&cd=2&cad=rja&uact=8&ved=0ahUKEwiktvKloKnUAhVPlxQKHbZXBmUQFggrMAE&url=http%3A%2F%2Fwww.springer.com%2Fcda%2Fcontent%2Fdocument%2Fcda_downloaddocument%2F9783987721972-c1.pdf%3FSGWID%3D0-0-45-466221-p173736563&usg=AFQjCNEHxgMqSGN4ErwCDcCdLGyU9cc9yQ&sig2=BienQ_HHCYOhHaWdih72iw

[2]HENG, Guo. Smart Cards and their Operating Systems [online]
Dostupné z: http://www.tml.tkk.fi/Studies/Tik-111.590/2001s/papers/heng_guo.pdf

[3] KUMAR, Sandeep S. *Elliptic Curve Cryptography for Constrained Devices: Algorithms, Architectures, and Practical Implementations.* Saarbruecken, Germany: VDM Verlag, 2008. ISBN 9783639068597.

[4] MALINA, Lukáš, BENEŠ, Vlastimil, HAJNÝ, Jan. Efficient and Secure Access Control System Based on Programmable Smart Cards. In 40th International Conference on Telecommunications and Signal Processing (TSP). 2016.

[5] MENEZES, Alfred, VAN OORSCHOT, Paul, VANSTONE, Scott. *Handbook of applied cryptography.* Boca Raton: CRC Press, 1997. 780 s. ISBN 0849385237.

[6]OCHODKOVÁ, Eliška. Matematické základy kryptografických algoritmů [online]
Dostupné z: http://www.cs.vsb.cz/ochodkova/courses/kpb/mzka.pdf

[7]RANKL, Wolfgang, EFFING, Wolfgang, Smart card handbook, Third edition, 2002

[8]GUILFOYLE, Tony. The ZeitControl BasicCard Family [online]
Dostupné z: http://www.smartcardbasics.com

[9] STALLINGS, William. *Cryptography and Network Security.* 4th edition. [s.l.] : [s.n.], 2006. 592 s. ISBN 0131873164.

[10]Transmition protocol, základní informace o protokolu T=0, T=1 [online]
Dostupné z: http://www.gorferay.com/the-t-0-transmission-protocol/

Seznam symbolů, veličin a zkratek

ROM	Read Only Memory
EEPROM	Electrically Erasable Programmable Read-Only Memory
AES	Advanced Encryption Standard
DES	Data encryption standard
APDU	Application protocol data unit
DSA	Digital signature algorithm
NR	Nyberg-Rueppel

Seznam příloh

A OBSAH PŘILOŽENÉHO CD

Na přiloženém CD se nachází vytvořený program. Program byl vytvořen ve vývojovém prostředí BasicCard Development Enviroment.